

[0079] When a search engine generates search results in response to a search query, a candidate document D that satisfies the query is assigned a query score, QueryScore, in accordance with the search query. This query score is then modulated by document D's page rank, PageRank, to generate a generic score, GenericScore, that is expressed as

$$\text{GenericScore} = \text{QueryScore} * \text{PageRank}.$$

[0080] This generic score may not appropriately reflect document D's importance to a particular user U if the user's interests or preferences are dramatically different from that of the random surfer. The relevance of document D to user U can be accurately characterized by a set of profile ranks, based on the correlation between document D's content and user U's term-based profile, herein called the TermScore, the correlation between one or more categories associated with document D and user U's category-based profile, herein called the CategoryScore, and the correlation between the URL and/or host of document D and user U's link-based profile, herein called the LinkScore. Therefore, document D may be assigned a personalized rank that is a function of both the document's generic score and the user profile scores. In one embodiment, this personalized score can be expressed as:

$$\text{PersonalizedScore} = \text{GenericScore} * (\text{TermScore} + \text{CategoryScore} + \text{LinkScore}).$$

[0081] FIGS. 9A and 9B represent two embodiments, both implemented in a network environment such as the network environment shown in FIG. 1. In the embodiment shown in FIG. 9A, the search engine 104 receives 910 via the front-end server 102, a search query from the client 118 that is submitted by a particular user. In response, the search engine 104 may optionally generate 915 a query strategy (e.g., the search query is normalized so as to be in proper form for further processing, and/or the search query may be modified in accordance with predefined criteria so as to automatically broaden or narrow the scope of the search query). The search engine 104 submits 920 the search query (or the query strategy, if one is generated) to the content server 106. The content server 106 identifies a list of documents that match the search query, each document having a generic score that depends on the document's page rank and the search query. This set of documents is also referred to as the search results, and they are typically ordered based on their GenericScore. In general, all the three operations are conducted by the search engine 104 and content server 106, which is on the server side of the network. There are two options on where to implement the operations following these first three steps.

[0082] In some embodiments that employ a server-side implementation, the user's ID is embedded in the query string provided by the client 118. This ID is passed from the front-end server 102 to the personalization server 108. Based on the user's ID, the user profile server 110 identifies 925 the user's user profile 230. The personalization server 108 analyzes each document in the search results to determine its relevance to the user's profile, creates 935 a profile score for the identified document. The profile score is based on any or all of the parts of the user profile 230 and then assigns 940 the document a personalized score that is a function of the document's generic and profile score. The personalization server 108 checks whether the current document is the last one of the search results. If not, the personalization server 108 processes the next document in

the search results. Otherwise, the search results are re-ordered 945 according to their personalized scores, to form the personalized search results. The personalized search results are provided to the front-end server 102 and to the content analysis module 112.

[0083] Embodiments using a client-side implementation are similar to the server-side implementation, except that after the search engine 104 obtains 920 the initial set of results, the search results sent to the corresponding client from whom the user submitted the query. This client stores the user's user profile 230 and it is responsible for re-ordering the documents based upon the user profile. In this embodiment, the client device has a local version of the personalization server 108, which performs essentially the same scoring and ranking functionality as previously described. Therefore, this client-side implementation may reduce the workload on the system 100. Further, since there is no privacy concern with the client-side implementation, a user may be more willing to provide private information to customize the search results. However, one limitation to the client-side implementation is that only a limited number of documents, e.g., the top 50 documents (as determined using the generic rank), may be sent to a client for reordering due to limited network bandwidth. In contrast, the server-side implementation may be able to apply a user's profile 230 to a much larger number of documents in the search result, e.g., 1000. Therefore, the client-side implementation may deprive a user access to those documents having relatively low generic ranks, but significantly personalized ranks.

[0084] FIG. 9B illustrates another embodiment. As before, the user's query and user ID is received via the front-end server 102, and the search engine 104 constructs 915 a generic query strategy. In addition, the search engine 104 adjusts 965 the generic query strategy according to the user's user profile 230 to create a personalized query strategy. This is done by the front-end server 102 providing the user's ID to the personalization server 108, which retrieves the user profile 230 and terms from the user's term profile 231. These terms are then added to the search query. The creation of the personalized query strategy can be performed either on the client side or on the server side of the system. This embodiment avoids the network bandwidth restriction facing the previous embodiment. The search engine 104 submits 970 the personalized query strategy to the content server 106. Since the content server 106 takes into account the additional personalized terms for the user's profile, the search results returned by the content server 106 have already been ordered 975 by the documents' personalized ranks.

[0085] The profiles 230 of a group of users with related interests may be combined together to form a group profile, or a single profile may be formed based on the documents identified by the users in the group. For instance, several family members may use the same computer to submit search queries to a search engine. If the computer is tagged with a single user identifier by the search engine, the "user" will be the entire family of users, and the user profile will be represent a combination or mixture of the search preferences of the various family members. An individual user in the group may optionally have a separate user profile that differentiates this user from other group members. In operation, the search results for a user in the group are ranked